

AMENDMENT TO THE CLAIMS

Please amend the claims as follows:

1. (Currently Amended) A memory management system ~~in a disc drive~~ having at least one data storage ~~disc~~medium, the memory management system comprising:

A₁ an arbitrated buffer memory having a plurality of memory address locations storing data associated with ~~logical block~~ addresses ~~on~~of the disc data storage medium; and
a traversal component configured to receive a requested traversal, arbitrate ownership of the ~~buffer~~-memory and to traverse sequentially mapped entries in the ~~buffer~~memory, associated with the requested traversal, prior to de-arbitrating itself from ownership of the ~~buffer~~-memory.

2. (Currently Amended) The memory management system of claim 1 wherein the traversal component comprises:

a memory accessing component sequentially accessing entries in the ~~buffer~~-memory based on the requested traversal and storing the entries in an accessing memory.

3. (Currently Amended) The memory management system of claim 2 wherein the traversal component comprises:

a traversal engine configured to access the entries in the accessing memory and determine whether the entries in the accessing memory correspond to ~~buffer~~-memory entries corresponding to the requested traversal.

4. (Currently Amended) The memory management system of claim 3 wherein the ~~buffer~~-memory comprises a linked list of memory locations.

5. (Currently Amended) The memory management system of claim 4 wherein the requested traversal includes a ~~buffer~~-memory starting address and a number of hops to take through the linked list beginning at the ~~buffer~~-memory starting address.

A1 6. (Currently Amended) The memory management system of claim 5 wherein the traversal engine is configured to determine whether the entries in the accessing memory correspond to ~~buffer~~-memory entries corresponding to the requested traversal by determining whether the entries in the accessing memory correspond to ~~buffer~~ memory locations in the linked list identified by the requested traversal.

7. (Original) The memory management system of claim 6 wherein the memory accessing component and the traversal engine are configured to operate substantially in parallel.

8. (Currently Amended) The memory management system of claim 7 wherein the traversal component is configured to voluntarily relinquish ownership of the ~~buffer~~-memory after traversing sequentially mapped entries in the ~~buffer~~-memory and to re-arbitrate for ownership of the ~~buffer~~-memory and continuing traversal of the ~~buffer~~-memory until the requested traversal is complete.

9. (Currently Amended) A method of managing a data buffer ~~in a disc drive~~, the method comprising:

- (a) receiving a traversal request to traverse the data buffer;
- (b) arbitrating for ownership of the data buffer; and
- (c) traversing all sequential entries in the data buffer, beginning at an entry point in the data buffer, corresponding to the traversal request prior to

voluntarily relinquishing ownership of the data buffer.

10. (Original) The method of claim 9 wherein receiving operation (a) comprises:

- (a) (1) receiving a data buffer starting address; and
- (a) (2) receiving a number of memory locations in the data buffer which must be made to complete the traversal request.

A,
11. (Original) The method of claim 10 wherein the data buffer comprises a linked list and wherein the traversing operation (c) comprises:

- (c) (1) reading adjacent entries in the data buffer into a register; and
- (c) (2) determining whether the entries in the register correspond to the traversal request.

12. (Original) The method of claim 11 wherein the traversing operation (c) further comprises:

performing the reading operation (c) (1) and the determining operation (c) (2) substantially in parallel.

13. (Original) The method of claim 12 wherein the traversing operation (c) further comprises:

- (c) (3) reducing the number of memory locations from the receiving operation (a) (2) by one each time the determining operation (c) (2) determines that an entry in the register corresponds to the traversal request.

14. (Original) The method of claim 13 and further comprising:

- (d) voluntarily relinquishing ownership of the data buffer after all sequential entries in the data buffer, corresponding to the traversal request, are read into

the register.

15. (Original) The method of claim 13 and further comprising:

- A₁
- (e) stopping the reading operation (c)(1) when it is determined in determining operation (c)(2) that an entry in the register does not correspond to the traversal request; and
 - (f) voluntarily relinquishing ownership of the data buffer.

16. (Original) The method of claim 15 and further comprising:

- (g) after ownership of the data buffer has been relinquished, determining whether the number of memory locations from the receiving operation (a)(2) has been reduced to zero.

17. (Original) The method of claim 16 and further comprising:

- (h) if in step (g) it is determined that the number of memory locations from the receiving operation (a)(2) has not been reduced to zero, re-arbitrating for ownership of the data buffer.

18. (Original) The method of claim 17 and further comprising:

- (i) continuing the traversing operation (c) until the number of memory locations to complete the traversal request is reduced to zero.

19. (Original) The method of claim 18 wherein the continuing operation (i) comprises:

- (i)(1) beginning traversing the data buffer at an entry point at a next data buffer location in the linked list corresponding to the traversal request.

20. (Currently Amended) A ~~disc-driven~~ data storage device,

comprising:

a ~~dis~~data storage medium; and

means for buffering data written to and read from the ~~dis~~
data storage medium by utilizing sequentially mapped
buffer data, associated with a requested traversal, to
decrease time associated with buffering.

A1